Prototype Parking Meter System – Phase 3

Project Plan

Team Code : May05-02

Client : Doug Houghton
Captain
Department of Public Safety
Iowa State University

Advisors : Prof. John Lamont, EE/CprE
Prof. Ralph Patterson III,
EE/CprE

Team Members : Chris Dasch, CprE
Jesse Pink, EE
Peter Stoltenow, CprE
Kwan Sin Wing Ted, CprE

Submission Date : 30 September 2004

REPORT DISCLAIMER NOTICE

DISCLAIMER: This document was developed as a part of the requirements of an electrical and computer engineering course at Iowa State University, Ames, Iowa. This document does not constitute a professional engineering design or a professional land surveying document. Although the information is intended to be accurate, the associated students, faculty, and Iowa State University make no claims, promises, or guarantees about the accuracy, completeness, quality, or adequacy of the information. The user of this document shall ensure that any such use does not violate any laws with regard to professional licensing and certification requirements. This use includes any work resulting from this student-prepared document that is required to be under the responsible charge of a licensed engineer or surveyor. This document is copyrighted by the students who produced this document and the associated faculty advisors. No part may be reproduced without the written permission of the senior design course coordinator.

Table of Contents

List of Figures ii
List of Tables iii
List of Symbols and Definitions iv
Abstract vi
Acknowledgement vii

1. Problem Statement 1
    1.1 General Problem Statement 1
    1.2 General Solution Approach 1
2. Operating Environment 3
3. Intended Users 3
4. Intended Uses 3
5. Assumptions 4
6. Limitations 4
7. Expected End Product and Other Deliverables 5
8. Proposed Approach 6
    8.1 Functional Requirements 6
    8.2 Constraints Considerations 6
    8.3 Technology Considerations 7
    8.4 Technical Approach Considerations 8
    8.5 Testing Requirements Considerations 9
    8.6 Security Considerations 9
    8.7 Safety Considerations 10
    8.8 Intellectual Property Considerations 10
    8.9 Commercialization Considerations 10
    8.10 Possible Risk and Risk Management 10
    8.11 Proposed Milestones and Evaluation Criteria 11
    8.12 Project Tracking Procedures 12
9. Statement of Works 14
    9.1 Task 1 – Project Familiarization 14
    9.2 Task 2 – Low-Level Slave Design 14
    9.3 Task 3 – Slave Implementation and Unit Testing 15
    9.4 Task 4 – System Integration and Testing 17
    9.5 Task 5 – On-Site Testing 18
    9.6 Task 6 – Bug Analysis and Removal 18
    9.7 Task 7 – Demonstration 18
    9.8 Task 8 – Documentation and Support 19
10. Estimated Resource Requirements 20
11. Schedules 22
12. Project Team Information 24
13. Closing Summary 25
14. References 26

List of Figures

Figure 1	:	Current system parking meter box	1
Figure 2	:	Parking lot sign	7
Figure 3	:	Customer additional time menu screen display	16
Figure 4	:	Change/print special event rate menu screen display	17
Figure 5	:	Change/print system parameters menu screen display	17
Figure 6	:	Gantt chart – Project deliverables	23
Figure 7	:	Gantt chart – Project tasks and subtasks	24

List of Tables

Table 1	:	Project milestones and importance	15
Table 2	:	Milestone evaluation	15
Table 3	:	Personnel effort requirements	19
Table 4	:	Other resource requirements	20
Table 5	:	Financial requirements	21

List of symbols and definitions

A **Assembly language**
A low-level computer language that consists of mnemonic codes and symbolic addresses corresponding to machine-language instructions

B

C **C**
A high-level object-oriented programming language

D **Dec04-02**
The senior design group responsible for the second phase of the project

DPS
Department of Public safety, the division of Iowa State University responsible for monitoring parking on the campus

E – F – G – H – J – K

L **LCD**
Liquid crystal display, a low-power digital display that uses liquid crystal cells that change reflectivity in an applied electronic field

Linux
An open source operating system based on UNIX traditionally used for servers.

M **May04-02**
The senior design group responsible for the first phase of the project

Motherboard
For this project, it is a main circuit board of the embedded computer through which all signals are directed.

MySQL
MySQL is a fast-relational database manager. A database manager enables adding, retrieving, and processing information stored in a database. The relational aspect of MySQL means that data is stored in separate tables rather than one large table. Relations between each table can be established and information can be retrieved using structured query language (SQL).

N – O – P – Q

R **RAM**

Random-access memory, the primary working memory in a computer used for the temporary storage of programs and data and in which the data can be accessed directly and modified

S **SQL**

A standardized language that approximates the structure of natural English for obtaining information from databases

T

U **USB**

Universal serial bus, a plug-and-play interface between a computer and peripheral devices, such as printers, modems, and keyboards

V

W **Window XP Embedded**

A compact and modular operating system provided by Microsoft Corporation that allows users to select the exact features and functionality of the OS required for the application or device.

Wired Ethernet

A trademark for a system for exchanging messages between computers on a local area network using wired coaxial, fiber optic or twisted-pair cables

X – Y – Z

Abstract

ISU currently has two pay-for-parking lots that have computerized control units with receipt printout capability. Each unit is programmable. The initial cost of each unit begins at $10,000 and rapidly escalates to more than $75,000 as features are added.

Working with the ISU Parking Division, the objective of this project would be to develop a demonstrable, microprocessor-based prototype unit with a number of features such as variable time-of-day rate, add-on time capability, etc as specified by the ISU Parking Division. The May04-02 and Dec04-02 project teams have developed an initial overall design. The design involves a dual-processor central unit and multiple user interface units. Working with the Dec04-02 team, the new project team will concentrate on the actual implementation and testing of the system. Figure 1 shows the current system, and the system being designed will be approximately the same in appearance.

Figure 1 – Current system parking meter box

Acknowledgement

The project team would like to thank the following people for their time, ideas and financial contributions to the project: Doug Houghton of the Department of Public Safety, Iowa State University and May04-02, Dec04-02 electrical / computer engineering senior design teams. We would also like to acknowledge our project advisors, Dr. John W. Lamont, and Professor Ralph Patterson III for their advice and guidance with the project.

1. Problem Statement

The following sections will provide a general overview of the problem to be addressed, and how this project will provide a solution for it.

1.1 General Problem Statement

Availability of parking on or near campus has become a concern at Iowa State University. Because of this, several pay-for-parking lots have been installed on the Iowa State campus. With traditional parking meters, one is needed for every parking spot. These lots use centralized units that are able to accept money and track multiple spaces from one or two locations. This setup provides advantages over the traditional parking meters, such as monitoring the entire lot and collecting money from one location.

However, there are several problems with the current system. The current units lack the ability to communicate with one another. This means that when the lot is checked for offenders, each machine must be checked individually. Also, if a user wishes to add time to a space they have already paid for, they must return to the same exact machine. Finally, the lack of communication means that if one unit is disabled, all the data stored in it is lost.

Another problem is the current system units are very difficult to program. DPS has requested the ability to program in university holidays, as well as change the hourly rates. The current units require that this be done by a specialist, a process that is both expensive and time consuming.

1.2 General Solution Approach

This project will attempt to solve this problem by providing a system to monitor the pay-for-parking lots. This system will be similar to the current pay-for-parking lots implemented on the Iowa State University campus, and in many ways improve on it. The new system will be more affordable and user-friendly, as well as easier to maintain.

The solution in development will be implemented with many units all of which will communicate with a central server. All units will be able to communicate using a master/slave solution, and users of the lot will be able to add time via any machine. The new system will allow DPS parking enforcement officers to receive a list of lot activity. In addition the system will have a redundant central processor and memory, which will create a much more robust solution that is currently available.

The new units will have a simple to use interface that will make it easier for people parking in the lots to use the system, and DPS to administer the system. Finally, the

system will be implemented with standard computer hardware. This will make duplication easy and decrease the cost of construction and maintenance of the units.

The May04-02 senior design team completed the first phase of the project. This group completed much of the initial design work. The current work will consist of two teams. The second group, Dec 04-02, has completed the design and will be responsible for building a master/server unit. The third phase will be competed by this team, May05-02, which will be responsible for implementing and testing the slave/client unit prototype.

2. Operating Environment

The unit will be installed in an outdoor location on the campus of Iowa State University in Ames, IA. It must be able to withstand all the weather conditions present in this location. The unit will be able to deal with both extremes of temperatures, as well as all forms of precipitation such as rain, snow, and hail.

The unit will be used on a regular basis, and often by users that may treat the unit roughly. Because of this, the unit must be durable and designed to withstand extended users.

Finally, because the unit will be located on a college campus, it must be sturdy, and resistant to attempts at vandalism.

3. Intended Users

Three classes of users will use the system. The first class includes college students at Iowa State University, faculty and staff of Iowa State University, and visitors to the Iowa State Campus.

The second class of users is DPS parking enforcement officers. They will need additional functionality in order to monitor the parking lots.

The third class of user is the administrator. This user will have access to all the features available to the previous class, in addition to change settings of the system, such as the rates.

4. Intended Uses

The system will have three classes of users (see section 3), each of which has different functions. The robustness of the system is too great to put a comprehensive list of all features, so below can be found a few examples of functions for each class of user.

- For the first class of users, people that park in the lot, the system will:
    - Allow parking spaces to be paid for by amount of time, or money
    - Allow time to be added to a parking space from any unit connected to the server unit
    - Print a hard-copy receipt if the user desires
- For the second class of users, parking enforcement, the system will:
    - Allow users to monitor paid and unpaid parking spots in the lot
    - Allow users to gather parking lot statistics
- In addition to the features above, for the third class of users the system will:
    - Allow users to change hourly rates
    - Allow users to set holidays
    - Allow users to add second class users

5. Assumptions

Below is a list of all the assumptions that we are taking into account when designing the system.

- The lot size will be no more the 1000 spaces
- The units will not provide change
- AC power will be provided to the unit
- The units will only except nickels, dimes, and quarters as payment
- Iowa State University Facilities Department will install the system

6. Limitations

Our system has a few functional requirements that must be taken into account when designing the system. Below lists a few of the limitations that either we set or that our customer had.

- The time to have the prototype completed is very limited
- The system must implement all the features of the current system
- The unit must withstand Iowa outdoor conditions
- The unit must be theft proof
- The user interface needs to be compact and easy to use
- The system must allow for different rates, depending on the time of day and holidays
- The hardware unit must print receipt upon request
- The server unit must consist of two redundant processors
- The server unit must have redundant storage
- The unit must be able to run for four hours or more if power goes out
- Users should be able to add time to their current remaining amount
- The hardware must provide the current payment status of the lot for parking enforcement

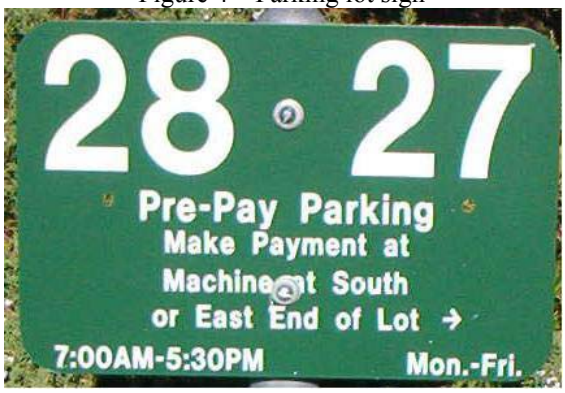
7. Expected End Result and Other Deliverables

The end products for this project will be a fully functional master/slave, multi-space parking meter system, complete user documentation, and a detailed technical specification. These items are detailed below:

Multi-space parking meter system

This system will consist of one of more slave computers connected to a central master computer. The system will be capable of handling up to 1000 parking spaces. The master unit will store all of the parking lot information. The slave unit(s) will retrieve this information and act as the interface from which parking time is purchased and administrators and/or supervisors can update pay rates, change parking schedules and manage a parking lot.

Figure 4 – Parking lot sign



User Manual

The user manual will be a document detailing the operation of all the machines in a non-technical way so that it can be understood by any user of the system. The operations described in the manual will include monitoring occupancy, making rate changes, as well as other enforcement functions. The final draft of the manual will be delivered with the system in December of 2004.

Technical Specification

The technical specification document will be a document describing the technical specifications of both the hardware and software running on the master and slave units. This document is not designed for common users, but instead is intended to be used as a resource for future developers and maintainers of the system. The completed technical specification of the system will also be delivered with the rest of the system in December 2004.

8. Proposed Approach

This section shall explain the planned method of completing the project; specifically, it will detail the completion of the slave unit(s) of the system, as the Dec04-02 group is responsible for the master and the master/slave protocol aspect of the project.

8.1 Functional Requirements

The following functions are required to successfully complete the slave unit of the project.

- Accepts client input/output
  The slave unit will have multiple ways of communicating with a user. Inputs include the keypad for customer and administrator/supervisor functions and the coin acceptor. Outputs include the receipt printer and a four line by forty character LCD display

- Communicates with master unit effectively and reliably
  The system will have a specialized protocol made specifically for communication between the master unit and all slaves. This unit will communicate over an Ethernet connection. Each slave will be able to poll the master unit in order to determine the appropriate course of action, such as time renewal or administrative functions.

- Offers a reasonably user-friendly user interface
  The system must provide a fairly intuitive interface for accessing the information that the system has to provide. This includes making it easy and straightforward for a paying customer to pay for and add time to a parking spot that he/she may be parked in. Also, the administrative functions must be easily accessible so that the administrators of the system can update and view information easily, saving both time and money.

8.2 Constraints Considerations

The entire project shall run under the following conditions and constraints.

- Weather resistance
  The entire system must be able to withstand all possible weather conditions that may occur on the Iowa State University campus. These include extreme heat and cold, precipitation, as well as severe winds (tornados) and associated debris.

- Weather resistance
  The entire system must be able to withstand all possible weather conditions that may occur on the Iowa State University campus. These include extreme heat and cold, precipitation, as well as severe winds (tornados) and associated debris.

- Durability
  The system must be durable, long lasting, and secure. Assuming the unit is built above ground, it must be able to withstand theft, vandalism, corrosion, and minor collisions with vehicles.

- Power requirements
  The master/slave system must be run off standard 110 AC power as well as being able to run off a battery backup should main power fail.

- Hardware requirements
  The master system must use redundant processing and storing capabilities to decrease chances of failure. The slave unit will only use a single processor and have considerably less storage than the master. Additionally, the slave unit will require the following pieces of equipment: LCD screen, coin acceptor, printer heater, printer, keypad, miniature computer case, and the external casing the user will see.

- Connectivity requirements
  The master/slave system must be able to complete all necessary communications over a standard Ethernet interface.

- Machine size requirements
  The control hardware will be placed inside one of the slave unit, so it must be large enough to facilitate installation, maintenance, and use as well as allow for the inclusion of all necessary components. It also must not be overly visually obtrusive.

8.3 Technology Considerations

Due to this being the final phase of development, most of the technology considerations have already been evaluated and decided upon.

- Master system hardware using dual processor-based system
  A system of this type is to utilize two separate processors that will allow the second processor to back up the first upon failure. It has redundant capability to store data to two sets of memory. If one processor fails, the second will take over automatically right where the first failed. This helps

alleviate downtime so the master control unit can process parking locations and continue to accept money.

- Programming language
  - MySQL

    The use of MySQL allows for easy creation of the central database and is easily integrated into both Linux and Windows XP Embedded operating systems.

  - C++

    Using the C language will allow for creation of a modular, robust, and easily modifiable software product. Many of the components of the system already have defined drivers for use in the C/C++ package.

- Communications hardware using wired Ethernet

  Wired (cat5/6) Ethernet allows for a more reliable, stable, and faster communication medium than other options that may be available, such as 801.11a/b/g.

8.4 Technical Approach Considerations

Like section 8.3, most of these considerations have already been decided upon. In order to create a multiple space parking meter system, a few things need to be implemented: the master unit's hardware and software, and the slave unit's hardware and software. The hardware must meet the needs of the client, which in this case is Iowa State's DPS. As such, the hardware will be an off the shelf x86 dual processor-based system. The system uses redundant processing and storing technology, specifically designed for the desktop and server market. This allows for increased modularity and reliability, easier software design and implementation, easier maintenance and better expandability.

The slave units will each be single processor units. They must be implemented in such a way that makes it easy to add components such as a printer, a keypad, a display, and another input device (the coin acceptor).

Finally, the unit requires a robust and feature-filled software package. In order to fulfill this requirement, C++ will be used for the creation of the software. Windows XP Embedded will be used as a development environment for the slave and Linux for the master unit. MySQL will be used for the database and should work well with both Linux and Windows.

8.5 Testing Requirements Considerations

The following is a definition of the methodologies and acceptance criteria to be used in the testing of the intermediate and end-products resulting from this project.

- Testing of hardware
    - The communications hardware needs to be able to operate properly in the wide range of temperatures and other environmental variables discussed above, communicating within the bounds of any error-checking and correction built into the software and without significant delays to make the system as responsive as possible.

    - The hardware as a whole will be deemed acceptable only if each individual piece operates under the aforementioned conditions. If any one type of hardware fails any of the tests, better precautions or different hardware must be selected to replace the unacceptable piece.

- Testing of software
    - The software needs to be tested to ensure that it functions as desired and is sufficiently robust to withstand daily use. Each separate module will first be tested individually; for example, we may first individually test the software module for the keypad and make sure that it is communicating with the computer correctly by inputting arbitrary input and checking to see if it is received correctly. After the separate testing phase of each component, we will put all of them together and make sure that each component will work together with the entire system as a whole. The May05-02 group as a whole will work together to do the above testing.

    - The last phase of software testing will be hooking up the slave and master units to make sure that the slaves are able to communicate effectively with the master units and vice versa and exchange the required information to offer the planned services. Both the May05-02 and the Dec04-02 group will be responsible for this testing.

    - The software will be deemed acceptable if it performs all of the desired functions without error and an uptime of seven days is observed.

8.6 Security Considerations

Security is always a concern when dealing with a system involving monetary transactions. An encryption protocol may be needed as data will be transmitted over a stand Ethernet link; however this has to be further evaluated since the Ethernet connection will not be connected to an external network. Each master and slave unit will be physically secure in the form of an enclosure designed to withstand vandalism,

theft, and a minor collision with a motor vehicle. This enclosure will have locking panels which will open up to the inside, offering access to the electronic hardware as well as the coin box. In order to make changes to the rate database, the administrator must have a unique pass code.

8.7 Safety Considerations

While the device inherently has very few safety concerns, it does contain electronic components, so electrical safety considerations must be considered and addressed throughout the design, construction, and use of the product.

8.8 Intellectual Property Considerations

Since this project introduces significant differences in design, construction, and capabilities when compared to other commercially available systems on the market. This should eliminate any trademark infringement issues. Any related works that are consulted in the course of this project will be acknowledged. Any intellectual property that results from this project will be the property of the team members and the Iowa State University Department of Public Safety.

8.9 Commercialization Considerations

When compared to popular models currently on the market, the system in development adds a considerable amount of new features and functionality while keeping costs drastically lower than any commercially available system. As a result, the probability of successfully marketing a product based upon the system currently in development is high.

8.10 Possible Risks and Risk Management

A number of potential risks have been identified:

- Code combination

  All code implementation will divide into several parts for each member. Even though we will have a plan to go by, it is possible that compatibility problems may occur when putting the code together due to different format and variables. Moreover, different writing style may lower the code readability and affect the correction time. Deciding on the variables and styles that we will be using ahead of time (before actually writing any code) will mitigate this risk. Also, any additional function or information must be reported to every member.

- Equipment damage

Unintended damage to project hardware components will take a toll in both time and replacements costs. Properly handling and storing all equipment will mitigate this risk.

- Equipment Availability

  Essential hardware may not be available for some unforeseen reason. Many of the items used are produced in low volume and are not always readily available. This risk will be mitigated by thoroughly researching hardware selections. The hardware that has been chosen will most likely provide the type of requirements that we need.

8.11 Project Proposed Milestones and Evaluation Criteria

- Design (Partially Completed)

  Proper completion of this milestone will result in the creation of a device design that meets the needs of the client and gets the approval of the faculty advisors. The hardware components have already been selected, so all that needs to be done for this phase is connecting the hardware together. This milestone will be considered complete when all the hardware is in place and running.

- Implementation

  The actual software coding of the parking meter system includes generating the code for various subsystems, including communication between devices, a database of rates, and generating statistical data. This milestone will be considered complete when every individual component is working together—only when every piece is working both independently and together will we consider this a success.

- Testing

  Verifying the code was correctly implemented, and that it meets the client's needs. This includes testing performance of the software as well as correctness of the code. The system will be tested in lab as well as in the field. As the system deals with monetary exchange, extensive testing will be required. While similar to the criteria for the implementation phase above, this will be evaluated by testing every function that has been implemented. This will be considered a success only when every function is working as designed.

- Demonstration

  Proper completion of this milestone will be the successful demonstration of the fully functional multi-space parking meter prototype unit. Product demos will be constructed during the design phase for validation purposes.

Table 1: Project Milestones and Overall Importance		
Milestone	**Importance**	
	Relative	**Percentage**
Problem Definition	Done	0
Research and technology	Done	0
Design	Low	10
Implementation	High	27
Testing	High	27
Demonstration	Medium	18
Documentation	Medium	18
TOTAL		100%

Table 2: Milestone Evaluation	
Evaluation Result	**Numerical Score**
Exceeded/Met	100%
Partially met	75%
Did not meet	50%
Did not attempt	0%

8.12 Project Proposed Tracking Procedures

The consistent flow of information regarding the status of the project is essential. The information will be provided in the form of status reports, schedule updates, and financial analysis.

The project plan contains a set of planned elements that are to be tracked and monitored throughout the duration of the project. The tracking activities include:

- Weekly status reports regarding current and planned activities will be sent out to the client, advisors and team members.

- The planned schedule will be compared to the actual progress to determine the current project status. If the actual progress has not met the planned progress, steps will be taken to determine the cause of the delay, the impact on the overall schedule, and to identify corrective measures to compensate for schedule variance.

- The planned budget will be compared to the actual expenditure by analyzing expenditures to date and estimating cost to and at completion.

These tracking measures will be used to provide critical project information without becoming an excessive burden. Microsoft Project will be used to help automate the tracking procedures and project management.

9. Statement of Work

The following is a formal statement of work containing a hierarchal list of major tasks and subtasks containing their objectives, approaches, and results. The first two tasks, Problem definition and Research and Technology, is already done in phase two.

9.1 Task 1 - Project Familiarization

Objective – The new team will gain an in depth understanding of the project as defined and high-level designed by previous teams.

Approach – Each team member will carefully read the Dec04-02 Project Plan and Software Functional Description (SFD) documents. Any questions the members have will be addressed by the faculty advisor(s) and earlier team members prior to the low-level design activity.

Expected Results – The team will be familiar with the overall design of the project so that they can better contribute to the low-level design activity.

9.2 Task 2 - Low-Level Slave Design

Objective – Devise a design for the slave component of the parking meter system using modular, object-oriented practices.

Approach – Several tasks will need to be completed during the design phase. First, user interface (UI) flowcharts will be constructed and/or studied for every section of the system. Second, descriptions of all necessary classes, data structures, global variables, and functions will be created from the flowcharts and SFD. Third, a block diagram illustrating the relationships between system objects and functions will be created.

Expected Results – All of the modules and objects of the system will be fully described before actual coding is implemented.

9.2.1 – User Interface Flowcharts

Objective – Create flowcharts of the user interface to ease later design and implementation efforts.

Approach – Convert each section of the SFD to a flowchart. Each section describes a menu with a finite number of choices for the user. The menu and result of each choice will be described graphically in the flowchart. Each decision point shall be enumerated with a state number.

Expected Results – A complete set of UI flowcharts that describe every possible user input string will be finished.

9.2.2 – Class and Function Descriptions

Objective – Every class, data structure, and function will be described in detail before implementation begins.

Approach – For each class, a list of attributes (including types and names) and operations (including return types, names, and parameters) shall be provided. For each data structure, a list of attributes will be provided. For each function, a function prototype will be written and a description of its parameters and result will be provided. Functions may be grouped into modules where appropriate.

Expected Results – Implementation will be simplified with all of the necessary classes and functions already described.

9.2.3 – Block Diagram

Objective – Relationships between functions and objects will be described graphically in a block diagram.

Approach – Using the material created in 2.1 and 2.2, a block diagram will be created that describes dependencies between system objects and functions. If function A is dependent on function B, an arrow points from A to B. If functions A and B are codependent, a line with arrows on each end connects them.

Expected Results – A graphical description of the entire software system is created.

9.3 Task 3 – Slave Implementation and Unit Testing

Objective – Create a single complete working prototype of the slave component will be created.

Approach – Software implementation of the slave will be split into three phases: First, each class will be coded and tested. Next, each function module will be coded and tested using the classes. Finally, the main module will integrate each of the function modules into a seamless user

interface. The hardware implementation of the slave will be completed by the EE major team member concurrently with software implementation.

Expected Results – The slave component will be completed and ready for testing.

9.3.1 – Class Implementation

Objective – Write and test code for each of the specified classes.

Approach – Classes will be divided amongst the CprE team members for coding. Each team member will be responsible for writing the class, and verifying that it meets the requirements specified in the function descriptions from 2.2.

Expected Results – Each class will be verified as working and ready for use in higher-level functions.

9.3.2 – Function Implementation

Objective – Write and test code for each of the specified functions.

Approach – Groups of functions will be divided amongst the CprE team members for coding. Each team member will be responsible for writing the set of functions, and verifying that they meet the requirements specified in the function descriptions from 2.2.

Expected Results – Each group of functions will be verified to meet the requirements from the design activity and to be ready for use with higher-level functions.

9.3.3 – Main Function Implementation

Objective – All of the functions will be integrated for use in a main function or set of functions.

Approach – The project leader will head an effort to integrate the tested modules into a complete working system. Other team members will assist the project leader as requested. Function stubs will be created to simulate responses from the master component.

Expected Results – A complete software package will be ready for testing.

9.3.4 – Hardware Implementation

Objective – Assemble the hardware components of the slave.

Approach – A single team member will be responsible for connecting all of the slave components together. He may get assistance from other team members or Dec04-02 team members as necessary. These components include: slave motherboard, slave case, power, coin acceptor, printer, liquid crystal display, and keypad.

Expected Results – The hardware for the system will be ready for software installation before the software is completed.

9.4 Task 4 – System Integration and Testing

Objective – The master and slave components will be connected and tested for correct communication and functionality.

Approach – Initially, the May05-02 group and the Dec04-02 group will work separately to ensure that their individual responsibilities work on their own. The May05-02 group will first test each individual component (i.e. the LCD, keypad, etc.) to ensure that they work. This will be done by designing simple routines that will perform simple communications between each component and will be monitored either by directing output to an external (from the system) monitor or (if working) the system's LCD output.

Next, all the components of the slave computer will be hooked up and tested extensively to ensure that all the components work together. This will be accomplished by designing a test routine that will test all the components together to ensure that all communications within the slave unit are working. During this phase, all functions that can be tested without being connected to the master/server part of the system will be tested. This includes all menu and logic functions. For functions that require a response from the server, the initial test version of the software will provide placebo information that will simulate the information that would otherwise come from the server, such as stall information, time, and rate structures.

The final phase of testing will come when both groups connect the slave and master computers together. During this phase, we will test every function again to ensure that all of the promised and desired features are working. This phase will include a manual programming of the rate database from scratch using one of the slave/client computers, followed by an actual testing period where we will simulate an actual parking lot

experience by putting in coinage to buy initial stall time and to renew time as well. We will simulate a second slave unit to ensure that rate and stall information can be updated from multiple slave units without causing conflicts in the system.

Expected Results – An integrated and tested prototype of the entire parking meter system, including one master and one slave component will be complete.

9.5 Task 5 – On-Site Testing

Objective – The complete system will be placed in a small test parking lot, and problems will be reported to the May05 team by DPS.

Approach – A bug reporting system will be devised by the project leader to provide a means for DPS to quickly and efficiently contact the team. Bugs will be given a priority by DPS, so that any bug that prevents the further operation of the system can be fixed as soon as possible. Non-critical bugs will be recorded for later analysis and fixing.

At least one member of the May05 team will be around during Thanksgiving and Winter breaks to provide support and troubleshooting should it be necessary.

Expected Results – By the end of the on-site test period, the vast majority of software defects will have been identified. Critical defects will have been fixed.

9.6 Task 6 – Bug Analysis and Removal

Objective – Remove any remaining bugs from the system.

Approach – A detailed list of software defects will have been compiled. The list will be split between team members for investigation and solution development. The team will meet to compare and implement the fixes, so that further defects will not be caused by bug fixes. Fixes will be split between group members and completed.

Expected Results – A defect-free system is completed and ready for shipment to DPS.

9.7 Task 7 – Demonstration

Objective – The final system will be demonstrated to DPS officials and faculty advisors at a specified date and time.

Approach – The May05-02 and the Dec04-02 groups will work together in order to develop an adequate demonstration that will explore many of the functions of all three of the user classes. The demonstration will include all commonly used functions as well as a few of the lesser-used ones. Showing all functions as described in the design document would take an unreasonable amount of time and as such is probably unfeasible.

Figures 3, 4, and 5 demonstrate some of the expected output screen of the parking meter that will be demonstrated.

Expected Results – The client will be satisfied that the system meets all specifications and requirements.

Figure 3 – Customer additional time menu screen display

```
'1' TO SPECIFY AMOUNT OF ADD'L TIME
'2' TO SPECIFY NEW EXPIRATION TIME
'3' TO BEGIN BY INSERTING COINS
'9' TO RETURN TO THE PREVIOUS MENU LEVEL
```

Figure 4 – Change/print special event rate menu screen display

```
'1' TO CHANGE SPECIAL-EVENT RATE VALUE
'2' TO PRINT SPECIAL-EVENT RATE VALUE
'3' TO RETURN TO PREVIOUS MENU LEVEL
```

Figure 5 – Change/print system parameters menu screen display

```
'2' TO CHANGE/PRINT RATE-SCHED. TIMES  ↑
'3' TO CHANGE/PRINT THE CALENDAR
'4' TO CHANGE/PRINT LOT SIZE
'5' TO CHANGE/PRINT DATE/TIME ITEMS    ↓
```

9.8 Task 8 – Documentation and Support

Objective – Develop documentation and support the parking meter.

Approach – The system will be fully documented for administrator and customer use. Also, the team will continue to support the system until May 2005.

Expected Results – A comprehensive document that explains all of the functions that an administrator and customer can do and how to do them and the support to help them do something that may be unclear.

10. Estimated Resources and Schedule

This section will provide an estimate of the resources required for the project and the project schedules. The effort that will be spent on the project has been divided into tasks as listed below.

- Task 1: Project Familiarization
- Task 2: Low-Level Slave Design
  - Subtask 2.1: User Interface Flowcharts
  - Subtask 2.2: Class and Function Descriptions
  - Subtask 2.3: Block Diagram
- Task 3: Slave Implementation and Unit Testing
  - Subtask 3.1: Class Implementation
  - Subtask 3.2: Function Implementation
  - Subtask 3.3: Main Function Implementation
  - Subtask 3.4: Hardware Implementation
- Task 4: System Integration and Testing
- Task 5: On-Site Testing
- Task 6: Bug Analysis and Removal
- Task 7: Demonstration

Table 3 – Personnel Effort Requirements

Personal Effort (in hours)

Name		Task 2			Task 3									**Total**
	Task 1	2.1	2.2	2.3	3.1	3.2	3.3	3.4	Task 4	Task 5	Task 6	Task 7	Task 8	
Christopher Dasch	10	4	11	9	9	38	35	2	19	17	17	5	15	**191**
Ted Kwan	8	3	7	7	7	41	27	2	17	17	20	5	12	**173**
Jesse Pink	18	9	4	6	3	31	30	8	24	18	13	5	17	**186**
Pete Stoltenow	14	11	7	6	6	37	35	2	20	14	13	5	16	**186**
Subtotal Hours	50	27	29	28	25	147	127	14	80	66	63	20	60	**736**
Total Hours	**50**	**84**			**313**				**80**	**66**	**63**	**20**	**60**	**736**

The estimated financial cost of equipment and other resources needed to create a working prototype for delivery are listed below. Note that this applies to the entire project, and not just this team. Many of the items have already been purchased.

Table 4 – Other Resource Requirements		
Equipment and Other Resources		
Item	**Team Hours**	**Cost**
Motherboard/Processor 1	0	$150.00
RAM 1	0	$50.00
Storage 1	0	$200.00
Motherboard/Processor 2	0	$150.00
RAM 2	0	$50.00
Storage 2	0	$200.00
LCD	0	$75.00
Keypad	0	$100.00
Misc. Buttons	0	$50.00
Printer Controller	0	$120.00
Ethernet Switch	0	$57.00
UPS Battery Backup Unit	0	$100.00
Housing	0	$0.00
Project Poster	10	$50.00
TOTAL	10	$1,352.00

Table 5 – Financial Requirements	
Financial Requirements	
Item	**Cost**
Parts and materials	$150.00
Motherboard/Processor 1	$50.00
RAM 1	$200.00
Storage 1	$150.00
Motherboard/Processor 2	$50.00
RAM 2	$200.00
Storage 2	$75.00
LCD	$100.00
Keypad	$50.00
Misc. Buttons	$120.00
Printer Interface	$57.00
Ethernet Switch	$100.00
UPS Battery Backup Unit	$100.00
Housing	$0.00
Project Poster	$50.00
Services	
Shipping and handling	$50.00
Binding	$30.00
Equipment subtotal	$1,532.00
Labor ($10.75 / hour)	
Christopher Dasch	$2,053.25
Ted Kwan	$1,892.00
Jesse Pink	$1,978.00
Pete Stoltenow	$1,988.75
Labor subtotal	$7,912.00
TOTAL (With Labor)	$9,444.00

11. Schedule

Since May05-02 team has inherited this project from two previous teams, the schedule for this team starts in the later design phase. Some of the project deliverables will therefore be delivered in mid-implementation, though they may need to completed at an accelerated pace.

Included below is a Gantt chart describing the expected project schedule over the next two semesters, with a complete prototype built and tested by the end of the first semester.

Figure 6 -- Gantt chart, Project deliverables

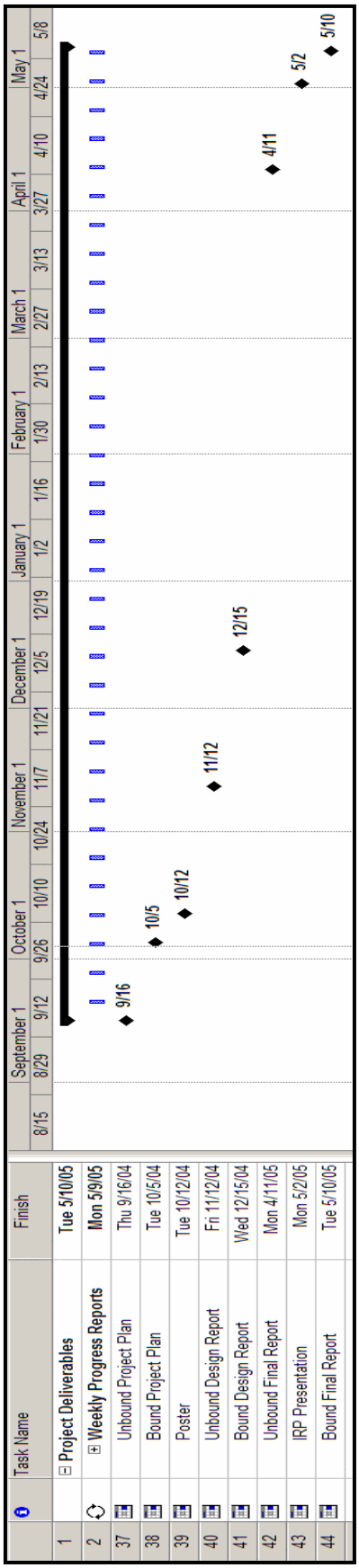
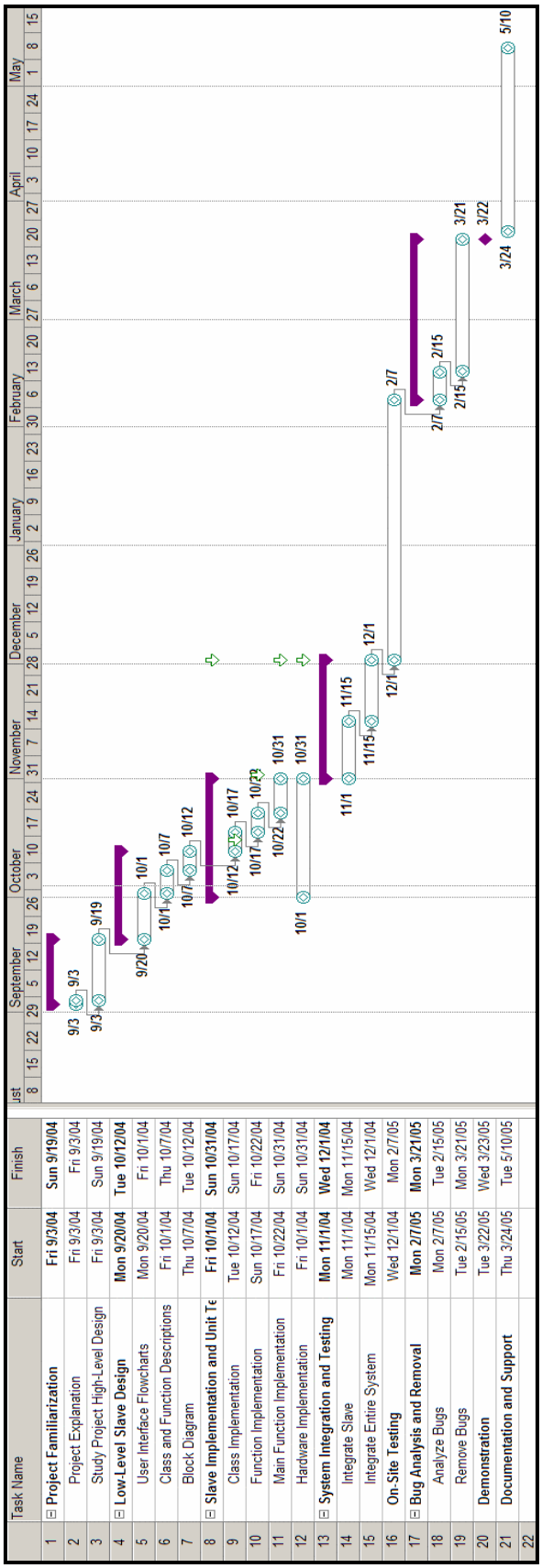


Figure 6 -- Gantt chart, Project tasks and subtasks



12. Project Team Information

The following is all the contact information for the client, faculty advisors, and the team members working on the project.

Client

Doug Houghton
Captain
Department of Public Safety
31 Armory Building
Ames, IA 50011
Vox: 515/294-1987
Fax: 515/294-0383
dad@iastate.edu

Faculty Advisors

Dr. John Lamont
324 Town Engineering
Iowa State University
Ames, IA 50010
Vox: 515/294-3600
Fax: 515/294-6760
jwlamont@iastate.edu

Professor Ralph Patterson III
326 Town Engineering
Iowa State University
Ames, IA 50010
Vox: 515/294-2428
Fax: 515/294-6790
repiii@iastate.edu

Team Members

Chris Dasch
218 S Walnut No 5
Ames, IA 50010
515/233-5189
nasch99@iastate.edu

Jesse Pink
311 Ash Ave
Ames, IA 50010
jlink7@iastate.edu

Peter Stoltenow
4226 Frederiksen Court
Ames, IA 50010
515/572-7860
pas@iastate.edu

Kwan Sin Wing Ted
1300 Gateway Hills #311
Ames, IA 50014
515/441-0224
kwanted@iastate.edu

13. Closing Summary

Parking is becoming an ever-increasing problem with more and more automobiles on the road. In densely populated areas such as urban centers and corporate and academic campuses, the problem is only worse. As Iowa State works to cope with its parking predicament, it also needs a cost-effective solution to finance the construction and maintenance of new lots. Current multi-space pay-for-parking systems are too expensive and rigid to meet the university's needs, and old-style individual parking meters are not practical for large lots. As a result, this project was created to devise and construct a system that can handle large parking areas, while at the same time being inexpensive enough to be used widely across campus and flexible enough to allow for diverse parking situations, and create a large source of revenue for Iowa State University.

14. References

Prototype Parking Metering System – Phase 2: Project Plan. - Dec 04-02
30 March 2004
From http://seniord.ee.iastate.edu/dec0402/Dec04-02%20Project%20Plan.pdf

Software Functional Description. – Dec 04-02, J. Lamont & R. Patterson III, 19 July 2004